ReportStream

Programmer's Guide for Organizations and Testing Facilities

VERSION 3.0 – February 2023

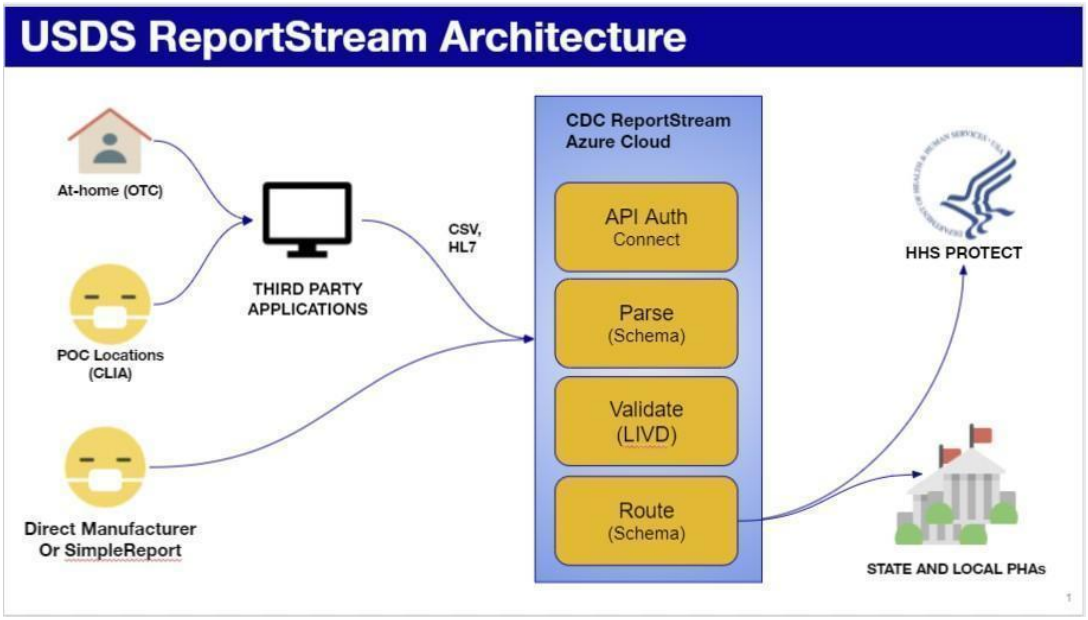
Table of contents

Introduction

About ReportStream

ReportStream is a free, open-source data platform that makes it easy for public health data to be transferred from testing facilities to public health departments.

ReportStream will automatically filter, transform, batch, and forward data to local, state, and federal jurisdictions based on both geographical and data quality filters provided by those jurisdictions.



ReportStream is not a permanent repository, EMR, or registry for health data. We only keep the data long enough to ensure it gets to the proper local, state, and federal jurisdictions.

About this guide

This programmer's guide enables those who are writing automated systems and tools to send laboratory and other health-related data to local, state, and federal jurisdictions. It helps you, the technical user at the testing facility or sending location, learn how to send data using the ReportStream Restful (REST) API.

Examples in this guide use curl commands for simplicity with the assumption you'll be coding these calls into your sending system. You can also use a program like Postman to test submissions.

About our API

The Waters API — the primary secure entry point to ReportStream — is named in memory of Dr. Michael Stephan Waters (1973-2020). His tireless work at the U.S. Food and Drug Administration championed diagnostic data interoperability efforts nationwide. ReportStream honors Dr. Waters through continuation and elevation of his work.

Release notes

Onboarding

Overview

1. **Kickoff**: As you get started, a ReportStream engineer will set up a kickoff call to review the process outlined in this guide and answer any questions.
2. **Validate and test your data**: There are three rounds of testing: validating formatting with fake data, testing the API connection with fake data, and testing PII data in production.
3. **Start sending your data**: Now that we know your files and connection will work, you can start sending your data through the API.

1. Kickoff

Before setting up your data, you will:

1. Attend a kickoff call with a ReportStream engineer
2. Get an Okta account
3. Agree to our Terms of Service

If you haven't connected with us yet, reach out so we can help you begin onboarding.

2. Validate and test your data

Test 1: Testing your formatting with fake data

To prepare your file for testing, review our data models and set up a sample file with fake data (artificially created, non-PII data). If you need fake data to use, reach out and we can provide that for you.

Currently, ReportStream can accept either a CSV file or HL7 input data. It's often easier to look at sample data in a file than to work from a schema. We can send you a file with fake data to review that will always successfully validate against the schema used to generate it. Note that because the data in those files are computer-generated, it may not feel realistic in some places.

View the data model and field requirements

Note for HL7 OTC Tests:

For this step, refer to the RADx MARS Getting started guide. Within that guide, you'll find information on field requirements, a tool outlining manufacturer-specific values, and a COVID-19 OTC-specific profile of the NIST HL7 v2 validator. To configure the validator, refer to NIST HL7v2 validator instructions at the bottom of the RADx MARS Getting started guide.

When you've formatted your fake (non-PII) data file, send your file to the ReportStream team. An engineer will work with you to correct any errors you receive.

Test 2: Testing your API connection in staging

After you have finalized a data model that works for you and ReportStream, the ReportStream team will begin onboarding you to our staging environment.

As part of the onboarding process, the ReportStream team will assign your unique client-id and set up your ReportStream account with the type of data you will be submitting. ReportStream will use the client-id to look up the associated data model and format (CSV, CSV OTC, or HL7) and validate the attached payload.

Your first step in this phase is to set up your authentication.

How to set up authentication

There are two methods of authenticating to ReportStream's REST API:

1. Token-based authentication with a public/private key pair
   Note: This method is the recommended best practice.
2. Using a shared secret API key

The examples below use the fake client-id healthy-labs, that you will change for your submissions. The examples submit the payload contained in the file `./healthy-labs-nonPII-data.csv (or .hl7)`. In the examples, data are submitted via an HTTP POST to the ReportStream staging system `reports` endpoint. The data submitted are sent as the payload of the POST, as is, with no changes.

Option 1: Example of token-based authentication with public/private key pair

Step 1: Prior to submission, send your public key to ReportStream.

1. Prior to connecting to the endpoint, you'll need a public/private keypair. There are many ways to do this. The steps below show how to create a key pair using `openssl`.

   EC

```
openssl ecparam -genkey -name secp384r1 -noout -out my-es-keypair.pem
openssl ec -in my-es-keypair.pem -pubout -out  my-es-public-key.pem
```

RSA

```
openssl genrsa -out my-rsa-keypair.pem 2048
openssl rsa -in my-rsa-keypair.pem -outform PEM -pubout -out my-rsa-public-key.pem
```

2. Send the public key to the ReportStream team (they'll associate it with your configuration within ReportStream). Once configured, continue with the steps below.

You only need to do this step once, not every time you submit reports. You can submit replacement keys to ReportStream at any time, following the steps above.

Step 2: At the time of submission, generate a signed JWT using your private key.

A JWT is a base64 encoded string that has three parts: `header`, `payload`, and `signature`.

You can find an example python program to generate a valid JWT on GitHub.

Here is an example, using the fake client-id healthy labs, of header and payload data that should appear in a ReportStream JWT, prior to signature:

```
{
   "header": {
      "kid": "healthy-labs.default",
      "typ": "JWT",
      "alg": "RS256"
   },
   "payload": {
      "iss": "healthy-labs.default",
      "sub": "healthy-labs.default",
      "aud": "staging.prime.cdc.gov",
      "exp": 1660737164,
      "jti": "4b713fcd-2514-4207-b310-620b95b749c5"
   }
}
```

Note:

- The exp (expiration time) should be a Unix time, five minutes after the time the token was generated.
- The jti (JWT ID) should be a random unique string, new with every call.
- Generate the signed JWT using your private key.

Step 3: Send the signed JWT to ReportStream to get a temporary bearer token

POST to the token URL, as in the example below, noting the following:

1. Use Content-Type: application/x-www-form-urlencoded.
2. In the `scope` parameter, replace the dummy string '`healthy-labs`' with your client-id, as assigned to you by ReportStream staff.
3. The `grant_type` and `client_assertion_type` parameters are always fixed values. The `grant_type` should be `client_credentials` and `client_assertion_type` should be `urn:ietf:params:oauth:client-assertion-type:jwt-bearer`, as in the example curl below.
4. In the client_assertion parameter, replace the <token-signing-secret> below with your JWT from above.
5. All the parameters are sent in the body/payload of the post (when using curl, via the `-d` option), not in the URL.

Here is an example 'curl' POST:

```
curl -X POST -H "Content-Type: application/x-www-form-urlencoded" -d
"scope=healthy-labs.default.report&grant_type=client_credentials&client_assertion_type=urn:ietf:
params:oauth:client-assertion-type:jwt-bearer&client_assertion=<token-signing-secret>"
"https://staging.prime.cdc.gov/api/token"
```

You should get something like this back, which will be valid for five minutes:

```
{"access_token":"<long-access-token>","token_type":"bearer","expires_in":300,"expires_at_seconds
":1625260982,"scope":"healthy-labs.default.report"}
```

Step 4: Submit data to ReportStream using the bearer token.

Use the access token returned above as the bearer token for the submission:

CSV example

```
curl -H "authorization:bearer <long-bearer-token>" -H "client:healthy-labs"  -H
"content-type:text/csv" --data-binary "@./healthy-labs-nonPII-data.csv"
"https://staging.prime.cdc.gov/api/waters"
```

HL7 example

```
curl -H "authorization:bearer <long-bearer-token>" -H "client:healthy-labs"  -H
"content-type:application/hl7-v2" --data-binary "@./healthy-labs-nonPII-data.hl7"
"https://staging.prime.cdc.gov/api/waters"
```

Again, always remember to replace the healthy-labs client-id with the client-id supplied to you by ReportStream staff.

Option 2: Example of shared secret key authorization

To use this method, you will need to create a Keybase account if you do not already have one.

Here's an example bash shell curl command submission to ReportStream using a shared secret API key. The example command submits the contents of the file './healthy-labs-nonPII-data.csv' to the endpoint using the client name healthy-labs. You'll use your own client-id.

The ReportStream team will provide you with the x-functions- key value for submissions to that client-id.

CSV example

```
curl -X POST -H "client:healthy-labs" -H "content-type:text/csv" –data-binary
"@./healthy-labs-nonPII- data.csv" -H "x-functions-key:<place-token-here>"
https://staging.prime.cdc.gov/api/reports
```

HL7 example

```
curl -X POST -H "client:super-labs" -H "content-type:application/hl7-v2" –data-binary
"@./super-labs-nonPII- data.hl7" -H "x-functions-key:<place-token-here>"
https://staging.prime.cdc.gov/api/report
```

Test in staging

Once authentication is complete, you can test your automation code as well as your code that handles responses using the staging API. Data is sent in the HTTP payload, either in CSV or HL7 2.5.1 format. You can use curl commands, Postman or another method of your choosing to post test submissions to the staging environment.

> **Note:** Do not send any PII or PHI to the staging system — only fake (dummy, example, synthetic) data is acceptable.

Let us know when you send submissions to the staging environment. We'll review that data and work with you to correct any issues. You may send as many fake data submissions to staging as is helpful.

For troubleshooting on your own, here is the complete endpoint input and response OpenAPI specification.

Test 3: Testing a file with PII in production

The ReportStream team will onboard you to the production system in training mode. If using a shared secret key, you'll receive API keys or tokens and the URL via Keybase. ReportStream won't forward or transport data received in training mode. However, the response message provides detailed information on where your data would have flowed if production mode was active.

3. Start sending your data

When you are ready, the ReportStream team will move you out of training mode and enable full production mode. Once in production, you can send a single record or up to 10,000 records in a single submission.

Data will automatically flow to appropriate state, local, and federal jurisdictional systems.

Note: Some jurisdictions require additional validation before sending data to their systems. If this affects your data submission, the ReportStream team will assist you in the process. Currently, the following states require additional validation:

- California
- Illinois
- New Jersey
- Washington

Responses from ReportStream

ReportStream responds to each API call with a response (JSON formatted) about the disposition of your data.

Errors and warnings

The ReportStream response may include warnings and/or errors based on validation of the submission against the expected format.

An accepted submission returns a 201 “httpStatus” code. Submissions with warnings but no errors will still be accepted. While ReportStream will accept a submission with warnings, we encourage you to fix those issues to ensure you will not have future problems sending to public health departments. However, one or more errors fail the entire submission.

Common validation errors

- Missing or mislabeled required columns/fields
- Missing or malformed data in required columns/fields
- CSV with “jagged” rows — differing number of columns within the payload
- Empty payload (an empty response is often a sign of a failed authorization, with a 401 response status). Make sure your token or key and the URL are correct.
- Incorrect client-id or other headers
- Incorrect data types (e.g. a character string when a numeric value is expected)

Common validation warnings

- Missing optional columns/fields

Note: There’s a flag allowing partial submissions. With this flag, successful elements in a batch will succeed, and the unsuccessful ones won’t. This flag requires extra code on your part to handle partial failures.

Response messages

Asynchronous processing

In most cases, we'll ask you to submit via ReportStream asynchronous (async) processing. This configuration setting is automatically enabled for users. Upon submitting data via ReportStream async processing, the REST endpoint returns almost immediately. However, ReportStream doesn't return information about where the tests will be sent.

In exchange for speed, the async submission response provides less initial information in the JSON. The initial response will provide errors and warnings, but no destination or filter information. The History Details API can be queried later to get full information about expected and actual destinations.

Example ReportStream response to an async submission:


```
{
  "submissionId":1604,
  "timestamp":"2022-02-10T13:50:19.162694Z",
  "sender":"simple_report.default",
  "httpStatus":201,
  "id":"3597ad7d-b92c-4bc0-a8fc-d909ed87bc90",
  "reportItemCount":2,
  "destinationCount":0,
  "destinations": [],
  "errors": [],
  "warnings": [],
  "topic":"covid-19",
  "warningCount":0,
  "errorCount":0
}
```

Synchronous processing

Example JSON response to a successful synchronous submission:


```
{
    "submissionId": 1588,
    "timestamp": "2022-02-09T16:59:33.789532Z",
    "sender": "simple_report",
    "reportItemCount": 2,
    "httpStatus": 201,
    "id": "e8880dcf-a201-4690-8e02-2871da739b61",
    "destinationCount": 2,
    "destinations": [
        {
            "organization_id": "de-dph",
            "service": "elr",
            "filteredReportRows": [],
            "sending_at": "2022-02-09T17:00:00.000000Z",
            "itemCount": 1,
            "sentReports": [ ],
            "organization": "Delaware Division of Public Health"
        },
        {
            "organization_id": "hi-phd",
            "service": "elr",
            "filteredReportRows": [],
            "sending_at": "2022-02-09T19:00:00.000000Z",
            "itemCount": 1,
            "sentReports": [ ],
            "organization": "Hawaii Public Health Department"
        }
    ],
    "errors": [],
    "warnings": [],
    "topic": "covid-19",
    "warningCount": 0,
    "errorCount": 0
}
```


Since the response is returned in real-time, the “destinations” section supplies information about where the submission is *expected* to be sent. The exact time that the public health department receives that data depends on the settings for that jurisdiction.

ReportStream features a History Details API that can be later queried to obtain the actual destinations and relevant detail. If you’d like to use this API, let the ReportStream team know, and we’ll provide you with additional information about requirements for Okta authentication.

The request is made with the submissionId in the earlier example.:

```
https://prime.cdc.gov/api/history/simple_report/submissions/1588
```

Response:


```
{
    "submissionId": 1588,
    "timestamp": "2022-02-09T16:59:33.789532Z",
    "sender": "simple_report",
    "reportItemCount": 2,
    "httpStatus": 201,
    "id": "e8880dcf-a201-4690-8e02-2871da739b61",
    "destinationCount": 2,
    "destinations": [
        {
            "organization_id": "de-dph",
            "service": "elr",
            "filteredReportRows": [],
            "sending_at": "2022-02-09T17:00:00.000000Z",
            "itemCount": 1,
            "sentReports": [
                {
                    "reportId": "38c84ec2-5741-4f2f-b234-25d774ec8caf",
                    "externalName":
"covid-19-43d64e18-ce56-482a-9134-f9f84a2c9d6f-20220209170000.hl7",
                    "createdAt": "2022-02-09T17:00:02.825148Z",
                    "itemCount": 1
                }
            ],
            "organization": "Delaware Division of Public Health"
        },
        {
            "organization_id": "hi-phd",
            "service": "elr",
            "filteredReportRows": [],
            "sending_at": "2022-02-09T17:00:00.000000Z",
            "itemCount": 1,
            "sentReports": [
                {
                    "reportId": "d9fae107-ef89-4fc0-b9b9-517219a4d2bb",
                    "externalName":
"covid-19-3560b0e8-c183-4132-ad0c-487a837f0e77-20220209170000.hl7",
                    "createdAt": "2022-02-09T17:00:02.822125Z",
                    "itemCount": 1
                }
            ],
            "organization": "Hawaii Public Health Department"
        }
    ],
    "errors": [],
    "warnings": [],
    "topic": "covid-19",
    "warningCount": 0,
    "errorCount": 0
}
```


The `sentReports` sections contain details about where and when the reports were transmitted.

JSON Error responses

In error cases, no report "id" UUID is returned, because no report was created based on the submission.

Example failure response and identical HistoryAPI response (Note the "id" is null, and the “httpStatus” is not 201):


```
{
    "submissionId": 1594,
    "timestamp": "2022-02-09T20:44:55.055545Z",
    "sender": "simple_report",
    "destinationCount" : 0,
    "httpStatus": 400,
    "id": null,
    "destinations": [],
    "errors": [
        {
            "scope": "item",
            "index": 1,
            "trackingId": "abcde",
            "type": "error",
            "message": "Blank value for element 'Patient_last_name' ('patient_last_name')"
        }
    ],
    "warnings": [],
    "topic": null,
    "warningCount": 0,
    "errorCount": 1
}
```


Example of a report level error:


```
{
    "submissionId": 1599,
    "timestamp": "2022-02-09T20:56:16.82117Z",
    "sender": "strac",
    "httpStatus": 400,
    "id": null,
    "destinationCount" : 0,
    "destinations": [],
    "errors": [
        {
            "scope": "report",
            "index": null,
            "trackingId": null,
            "type": "error",
            "message": "CSV file has an inconsistent number of columns on row: 3"
        }
    ],
    "warnings": [],
    "topic": null,
    "warningCount": 0,
    "errorCount": 1
}
```

Data model

API CSV and HL7 Field Requirements

Contents:

Legend:

Yes: Required field for acceptance

Yes – Conditional: Required, but only under certain circumstances. Review the field's Data Requirements and Additional Guidance for more information.

Requested: Field should be populated if available. In addition, some states may treat this as a required field.

No: Not a hard requirement. In the interest of providing complete information to public health agencies, please populate the field if data is available. For CSV implementations please provide the CSV column header even if no data is entered.

Note about OTC reporting:

- For CSV implementations, include only the columns marked with "(OTC)" in the CSV Column Names below.
- For HL7 implementations, do not use this table. Refer to the RADx MARS OTC information in 2. Validate and test your data.

Common errors:

- **Two of the most important and often overlooked pieces of required data are the deviceIdentifier (OBX-17.1) and testPerformed (OBX-3.1).** These fields must match exactly to the appropriate row in the LOINC In Vitro Diagnostic (LIVD) test code mapping. This is the most updated LIVD mapping from the CDC. Specifics about each field are detailed in the tables below.

- **The preferred timestamp formatting for CSV and HL7 is yyyyMMddhhmmss+/-zzzz.** If the UTC offset (+/-zzzz) is not present, results should be normalized to a single time zone that's agreed upon during the onboarding process.

Patient data elements

CSV Column Names	HL7 Field / Component	Fed Required?	State Required?	Data Requirements	Additional Guidance
patient_id (OTC)	PID-3.1	No	Requested	**Optional - Requested:** Enter unique patient identifier. This is typically the Medical Record Number. ***Do not send a Social Security Number.***	This value is optional and can be left blank if no information is provided. *Some jurisdictions may require this field, ReportStream will notify you if this is the case.*
patient_last_name (OTC)	PID-5.1	No	Yes	Enter patient's last name.	File will fail if field left blank.
patient_first_name (OTC)	PID-5.2	No	Yes	Enter patient's first name.	File will fail if field left blank.
patient_name_middle (OTC)	PID-5.3	No	No	**Optional:** Enter patient's middle name, if known.	This value is optional and can be left blank if no information is provided.
patient_street (OTC)	PID-11.1	No	Yes	Enter patient's home address.	File will fail if field left blank. If no address given or homeless, populate this field with ** Unknown / Not Given ** or ** Homeless **.
patient_street2 (OTC)	PID-11.2	No	No	**Optional:** Enter patient's additional address information, if applicable.	This value is optional and can be left blank if no information is provided.
patient_city (OTC)	PID-11.3	No	Yes	Enter patient's city.	File will fail if field left blank. If no city given or homeless, populate this field with the ordering facility information.
patient_state (OTC)	PID-11.4	No	Yes	Enter patient's state using the two-character abbreviation.	File will fail if field left blank. If no state given or homeless, populate this field with the ordering facility information.
patient_county (OTC)	PID-11.9	Yes	Yes	Enter patient's county/parish name.	Required for reporting to certain jurisdictions.
patient_zip_code (OTC)	PID-11.5	Yes	Yes	Enter patient's zip code. **Accepted Format:** 12345 12345-6789	File will fail if value is not entered using acceptable format or field is left blank. If no zip code given or homeless, populate field with the ordering facility information.

patient_phone_number (OTC)	PID-13.7	No	Yes - Conditional	Enter patient's phone number, if known. **Accepted Format:** 000-000-000	If no phone number given or homeless, populate field with the ordering facility information.
patient_dob (OTC)	PID-7.1	No	Yes	Enter patient's date of birth. **Accepted Format:** yyyyMMdd	File will fail if value is not entered using accepted format or field is left blank.
patient_gender (OTC)	PID-8.1	Yes	Yes	Enter patient's gender. **Accepted Values (HL70001):** • M or Male • F or Female • O or Other • U or Unknown • A or Ambiguous • N or Not applicable	File will fail if value not entered using accepted values or field is left blank. Accepted values come from values mapped to LOINC codes you can find in the PHIN VADS system.
patient_race (OTC)	PID-10.1	Yes	Yes	Enter patient's race. **Accepted Values (HL70005):** • 1002-5 or American Indian or Alaska Native • 2028-9 or Asian • 2054-5 or Black or African American • 2076-8 or Native Hawaiian or Other Pacific Islander • 2106-3 or White • 2131-1 or Other • ASKU or Ask, but unknown • UNK or Unknown	File will fail if numeric values or text values are not entered using acceptable values or field is left blank. Accepted values come from values mapped to LOINC codes you can find in the PHIN VADS system.
patient_ethnicity (OTC)	PID-22.1	Yes	Yes	Enter patient's ethnicity. **Accepted Values:** • 2135-2 or H or Hispanic or Latino • 2186-5 or N or Not Hispanic or Latino • UNK or U or Unknown	File will fail if numeric values or text values are not entered using acceptable values or field is left blank. Accepted values come from values mapped to LOINC codes you can find in the PHIN VADS system.
patient_preferred_language	PID-15	No	No	**Optional:** Enter patient's preferred language, if known. **Example Accepted Values:** eng OR English spa OR Spanish fre OR French jpn OR Japanese	Use the Concept Code or Concept Name from the PHIN VADS ISO-639 table, which can be found at https://phinvads.cdc.gov/vads/ViewValueSet.action?id=D0858308-9AB3-EA11-818F-005056ABE2F0#.
patient_email (OTC)	PID-13.4	No	No	**Optional:** Enter patient's email address, if known. **Accepted Value:** Numeric or text	This value is optional and can be left blank if no information is provided.

Order and result data elements

CSV Column Names	HL7 Field / Component	Fed Required?	State Required?	Data Requirements	Additional Guidance
accession_number (OTC)	ORC-3.1 OBR-3.1 SPM-2.2 MSH-10 ORC-2.1 OBR-2.1	Yes	Yes		An accession number is a unique ID that identifies a single result. This field is important for public health departments to refer back to a test event. File will fail if field left blank. For OTC a value is generated by ReportStream based on the patient id.
equipment_model_name (OTC)	OBX-17.2	Yes	Yes	Enter equipment model name value from Department of Health and Human Services' (HHS) LOINC Mapping spreadsheet. **Examples:** 1) "ID NOW" 2) "BD Veritor System for Rapid Detection of SARS-CoV-2*" 3) "BD Veritor System for Rapid Detection of SARS-CoV-2 & Flu A+B*" 4) "RightSign COVID-19 IgG/IgM Rapid Test Cassette*"	File will fail if value not entered using accepted values or field is left blank. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it. Click on the "LOINC Mapping" tab. Go to Column B, labeled "Model" to locate the corresponding value to enter.
n/a	OBX-17.1	Yes	Yes	Enter device identifier from Department of Health and Human Services' (HHS) LOINC Mapping spreadsheet. **Examples:** 1) "10811877011269" 2) "BD Veritor System for Rapid Detection of SARS-CoV-2_Becton, Dickinson and Company (BD)" 3) "BD Veritor System for Rapid Detection of SARS-CoV-2 & Flu A+B_Becton, Dickinson	File will fail if value not entered using accepted values or field is left blank. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it.

				and Company (BD)" 4) "RightSign COVID-19 IgG/IgM Rapid Test Cassette_Hangzhou Biotest Biotech Co., Ltd."	Click on the "LOINC Mapping" tab. Go to Column M, labeled "Testkit Name ID" to locate the corresponding value to enter.
equipment_uid (OTC)	OBX-18.1	Yes	Yes	Enter Equipment UID information from Department of Health and Human Services' (HHS) LOINC Mapping spreadsheet. **Examples:** 1) "00811877010616" 2) "BD Veritor Plus System_Becton Dickinson" 3) "BD Veritor Plus System_Beckton Dickinson" 4) "(Test kit device)"	File will fail if value not entered using accepted values or field is left blank. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it. Click on the "LOINC Mapping" tab. Go to Column O, labeled "Equipment UID" to locate the corresponding value to enter.
test_performed_code (OTC)	OBX-3.1	Yes	Yes	Enter TestPerformed Code value from Department of Health and Human Services' (HHS) LOINC Mapping spreadsheet. **Examples:** 1) "94534-5" 2) "94558-4" 3) "97097-0" 4) "94507-1" "94508-9"	File will fail if value not entered using acceptable values or field is left blank. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it. Click on the "LOINC Mapping" tab. Go to Column F, labeled "Test Performed LOINC Code". Locate the corresponding value to enter.
test_result (OTC)	OBX-5.1	Yes	Yes	Enter a numeric SNOMED code (preferred) or common text value listed. **Examples:** "260373001"	File will fail if value is not entered using accepted text values or SNOMED codes, or if the field is left blank.

				“Positive” “Negative” “Not Detected” “Detected” “Invalid Result”	Enter a value from the common values listed. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it. Click on the "LOINC Mapping" tab. Go to Column E, labeled "Vendor Result Description". Locate SNOMED code value and enter into field (Example: Positive = 260373001).
order_test_date (OTC)	ORC-15.1	Yes	Yes	Enter test ordered date. **Accepted Format:** yyyyMMddhhmmss+/-zzzz	File will fail if value is not entered using acceptable format or field is left blank.
specimen_collection_date (OTC)	SPM-17.1	Yes	Yes	Enter specimen collection date. **AcceptedFormat:** yyyyMMddhhmmss+/-zzzz	If unknown, populate field with the order_test_date value. In most cases, these are the same. Can be left blank for CSV if same as order_test_date.
testing_lab_specimen_received_date (OTC)	SPM-18.1	No	No	Enter testing lab specimen received date. **AcceptedFormat:** yyyyMMddhhmmss+/-zzzz	If unknown, populate field with the order_test_date value. In most cases, these are the same. Can be left blank for CSV if same as order_test_date.
test_result_date (OTC)	OBX-14.1	Yes	Yes	Enter test result date. **Accepted Format:** yyyyMMddhhmmss+/-zzzz	File will fail if value is not entered using acceptable format or field is left blank.
date_result_released (OTC)	OBR-22	Yes	Yes	Enter test report date. **Accepted Format:** yyyyMMddhhmmss+/-zzzz	File will fail if value is not entered using acceptable format or field is left blank.
comment	NTE-3	No	No	Any comments from a physician or lab technician you want to relay to your public health department can be entered here. This field is not intended for	This value is optional and can be left blank if no information is provided. *Do not include commas (,) in any CSV comments unless the field is*

				characteristics of COVID-19 tests or statements about false positive or negative results.	*encapsulated in quotes (").*
test_result_status (OTC)	OBX-11 OBR-25	Yes	Yes	**Accepted Values:** ● "F" for Final Result. ● "C" for Corrected Result	Enter test result status using the accepted format. If left blank, value will default to "F" for CSV.

Specimen data elements

CSV Column Names	**HL7 Field / Component**	**Fed Required?**	**State Required?**	**Data Requirements**	**Additional Guidance**
specimen_type (OTC)	SPM-4	Yes	Yes	Enter a numeric SNOMED code (preferred) or common text value listed. **Examples:** "697989009" "Nasal Swab" "Nasopharyngeal Swab" "Anterior Nares Swab" "Throat Swab" "Oropharyngeal Swab" "Whole Blood" "Plasma" "Serum"	File will fail if value not entered using acceptable text values or SNOMED codes or field is left blank. Go to https://www.cdc.gov/csels/dls/sars-cov-2-livd-codes.html. Click on the Mapping Tool labeled "LIVD SARS-CoV-2 Test Codes.xlsx" to download the file. Locate the saved file on your computer and open it. Click on the "LOINC Mapping" tab. Go to Column D, labeled "Vendor Specimen Description". Locate the corresponding text value or SNOMED code and enter into field (example: Anterior Nares Swab = "697989009").
n/a	SPM-8	Requested	Requested	Enter a numeric SNOMED code for the specimen source site code.	For CSV, this is populated by ReportStream based on the specimen_type value.

Ordering provider data elements

CSV Column Names	HL7 Field / Component	Fed Required?	State Required?	Data Requirements	Additional Guidance
ordering_provider_id (OTC)	ORC-12.1	Yes	Yes	Enter National Provider Identifier (NPI). ReportStream prefers this value, however if NPI is unknown enter local coding. **Examples:** • NPI example: 1013012657 • Local code example: muc1290	NPI is a 10-character all-numeric identification number to uniquely identify a health care provider. NPIs can be found at https://npiregistry.cms.hhs.gov/. *Some jurisdictions may not accept a local code, ReportStream will work with you if this is the case*. This field may be left blank for OTC tests.
ordering_provider_last_name (OTC)	ORC-12.2	No	Yes	Enter the last name of the ordering provider.	File will fail if field left blank for non-OTC tests. May be left blank for OTC.
ordering_provider_middle_name (OTC)	ORC-12.4	No	No	**Optional:** Enter ordering provider's middle name, if known.	This value is optional and can be left blank if no information is provided.
ordering_provider_first_name (OTC)	ORC-12.3	No	Yes	Enter the first name of the ordering provider.	File will fail if field left blank for non-OTC tests. May be left blank for OTC.
ordering_provider_street (OTC)	ORC-24.1	Requested	Yes	Enter the street address of the ordering provider.	File will fail if field left blank for non-OTC tests. May be left blank for OTC.
ordering_provider_street2 (OTC)	ORC-24.2	No	No	**Optional:** Enter ordering provider's additional address information, if applicable.	This value is optional and can be left blank if no information is provided.
ordering_provider_city (OTC)	ORC-24.3	Requested	Yes	Enter ordering provider's city.	File will fail if field left blank for non-OTC tests. May be left blank for OTC.
ordering_provider_state (OTC)	ORC-24.4	Requested	Yes	Enter ordering provider's state using the two-character abbreviation.	File will fail if field left blank for non-OTC tests. May be left blank for OTC.
ordering_provider_zip_code (OTC)	ORC-24.5	Requested	Yes	Enter ordering provider zip code. **Accepted Format:**	File will fail if value is not entered using accepted format or field is left blank.

				12345 12345-6789	
ordering_provider_phone_number (OTC)	ORC-14.7	Requested	Yes - Conditional	Enter ordering provider's phone number. **Accepted Format:** 000-000-000	File will fail if value is not entered using accepted format or field is left blank.

Testing facility data elements

CSV Column Names	**HL7 Field / Component**	**Fed Required?**	**State Required?**	**Data Requirements**	**Additional Guidance**
testing_lab_clia (OTC)	OBX-23.10	Yes	Yes	Enter testing facility's CLIA number.	File will fail if left blank. CLIA numbers can be found at https://www.cdc.gov/clia/LabSearch.html. For OTC, use 00Z0000014 For prescription, use 00Z0000015
testing_lab_name	OBX-23.1	No	Yes	Enter testing facility's name.	File will fail if field left blank.
testing_lab_street	OBX-24.1	No	Yes	Enter the street address of the testing facility.	File will fail if field left blank.
testing_lab_street2	OBX-24.2	No	No	**Optional:** Enter testing facility's additional address information, if applicable.	This value is optional and can be left blank if no information is provided.
testing_lab_city	OBX-24.3	No	Yes	Enter testing facility's city.	File will fail if field left blank.
testing_lab_state	OBX-24.4	Yes	Yes	Enter testing facility's state using the two-character abbreviation.	File will fail if field left blank.
testing_lab_zip_code	OBX-24.5	Yes	Yes	Enter testing facility's zip code. **Accepted Format:** 12345 12345-6789	File will fail if value is not entered using accepted format or field is left blank.
testing_lab_phone_number	N/A	No	No	Enter testing lab's phone number, if known. **Accepted Format:** • 000-000-0000	File will fail if value is not entered using accepted format.

Ask-On-Entry (AOEs)

CSV Column Names	HL7 Field / Component	Fed Required?	State Required?	Data Requirements	Additional Guidance
pregnant	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's pregnancy status. **OBX-3.1** 82810-3 **OBX-5.1** 77386006 (Yes) 60001007 (No) 261665006 (Unknown) **CSV** Y (Yes) N (No) U (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
employed_in_healthcare	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's employment in healthcare status. **OBX-3.1** 95418-0 **OBX-5.1/CSV** Y (Yes) N (No) U (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
symptomatic_for_disease (OTC)	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's symptomatic for disease status. **OBX-3.1** 95419-8 **OBX-5.1/CSV** Y (Yes) N (No) U (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
illness_onset_date (OTC)	OBX-3.1 OBX-5.1	Requested	Requested	Enter patient's illness onset date. **OBX-3.1** 65222-2 **OBX-5.1 Accepted Format:** yyyyMMdd	Field is not required, but requested for thorough reporting.

resident_congregate_setting	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's congregate housing status. **OBX-3.1** 95421-4 **OBX-5.1/CSV** Y (Yes) N (No) U (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
residence_type	N/A	Requested	Requested	**Optional - Requested:** Enter the type of facility providing care for patient. **Accepted Values:** 22232009 (Hospital) 2081004 (Hospital ship) 32074000 (Long Term Care Hospital) 224929004 (Secure Hospital) 42665001 (Nursing Home) 30629002 (Retirement Home) 74056004 (Orphanage) 722173008 (Prison-based care site) 20078004 (Substance Abuse Treatment Center) 257573002 (Boarding House) 224683003 (Military Accommodation) 284546000 (Hospice) 257628001 (Hostel) 310207003 (Sheltered Housing) 57656006 (Penal Institution) 285113009 (Religious institutional residence) 285141008 (Work environment) 32911000 (Homeless) 261665006 (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
hospitalized	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's hospitalization status. **OBX-3.1** 77974-4 **OBX-5.1/CSV** Y (Yes) N (No) U (Unknown)	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.
icu	OBX-3.1 OBX-5.1	Requested	Requested	**Optional - Requested:** Enter patient's intensive care unit (ICU) status. **OBX-3.1**	Field is not required, but requested for thorough reporting. Enter one of the acceptable values exactly as displayed.

				77974-4 **OBX-5.1/CSV** Y (Yes) N (No) U (Unknown)	

Reporting and ordering facility data elements

All Reporting and Ordering Facility Data Elements can be left blank for CSV if same as Testing Facility Data Elements

CSV Column Names	**HL7 Field / Component**	**Fed Required?**	**State Required?**	**Data Requirements**	**Additional Guidance**
reporting_facility_name	MSH-4.1	Yes	Yes		Can be left blank for CSV if same as testing_lab_name.
reporting_facility_clia	MSH-4.2	Yes	Yes		Can be left blank for CSV if same as testing_lab_clia.
ordering_facility_name	ORC-21.1	No	Yes – Conditional	Enter ordering facility name.	Can be left blank for CSV if same as testing_lab_name.
ordering_facility_street	ORC-22.1	No	Yes – Conditional	Enter the street address of the ordering facility.	Can be left blank for CSV if same as testing_lab_street.
ordering_facility_street2	ORC-22.2	No	No	**Optional:** Enter ordering facility's additional address information, if applicable.	This value is optional and can be left blank if no information is provided.
ordering_facility_city	ORC-22.3	No	Yes – Conditional	Enter ordering facility's city.	Can be left blank for CSV if same as testing_lab_city.
ordering_facility_state	ORC-22.4	No	Yes – Conditional	Enter ordering facility's state using the two-character abbreviation.	Can be left blank for CSV if same as testing_lab_state.
ordering_facility_zip_code	ORC-22.5	No	Yes – Conditional	Enter ordering facility zip code. **Accepted Format:** • 12345 • 12345-6789	Can be left blank for CSV if same as testing_lab_zip_code.
ordering_facility_phone_number	ORC-23	No	Yes – Conditional	Enter ordering facility phone number. **Accepted Format:** 000-000-0000	Can be left blank for CSV if same as testing_lab_phone_number.

Sample payloads and output

Sample CSV payload and output

Input

Download a sample file

Response


```
{
  "id" : "dbfbb65f-f6f0-4d85-a723-32b63283f068",
  "submissionId" : 2412,
  "overallStatus" : "Waiting to Deliver",
  "timestamp" : "2022-05-16T14:39:02.159Z",
  "plannedCompletionAt" : "2022-05-16T14:40:00.000Z",
  "actualCompletionAt" : null,
  "sender" : "csvuploadertest.default",
  "reportItemCount" : 5,
  "errorCount" : 0,
  "warningCount" : 0,
  "httpStatus" : 201,
  "destinations" : [ {
    "organization" : "Alabama Public Health Department",
    "organization_id" : "al-phd",
    "service" : "elr",
    "itemCount" : 5,
    "itemCountBeforeQualityFiltering" : 5,
    "sending_at" : "2022-05-16T14:40:00.000Z",
    "filteredReportRows" : [ ],
    "filteredReportItems" : [ ],
    "sentReports" : [ ],
    "downloadedReports" : [ ]
  } ],
  "errors" : [ ],
  "warnings" : [ ],
  "topic" : "covid-19",
  "externalName" : null,
  "destinationCount" : 1
}
```

Sample HL7 2.5.1 payload and output

Input

FHS|^~\&|CDC PRIME - Atlanta,^2.16.840.1.114222.4.1.237821^ISO|CDC PRIME -
Atlanta,^2.16.840.1.114222.4.1.237821^ISO|||202108031315+0000
BHS|^~\&|CDC PRIME - Atlanta,^2.16.840.1.114222.4.1.237821^ISO|CDC PRIME -
Atlanta,^2.16.840.1.114222.4.1.237821^ISO|||202108031315+0000
MSH|^~\&|CDC PRIME - Atlanta,^2.16.840.1.114222.4.1.237821^ISO|Winchester House^05D2222542^ISO|CDPH CA
REDIE^2.16.840.1.114222.4.3.3.10.1.1^ISO|CDPH_CID^2.16.840.1.114222.4.1.214104^ISO|20210803131511.0147+0000||O
RU^R01^ORU_R01|1234d1d1-95fe-462c-8ac6-46728dba581c|P|2.5.1|||NE|NE|USA|UNICODE
UTF-8|||PHLabReport-NoAck^ELR_Receiver^2.16.840.1.113883.9.11^ISO
SFT|Centers for Disease Control and Prevention|0.1-SNAPSHOT|PRIME Data Hub|0.1-SNAPSHOT||202107260000
PID|1||09d12345-0987-1234-1234-111b1ee0879f^^^Winchester
House&05D2222542&ISO^PI^&05D2222542&ISO||Bunny^Bugs^C^^^^L||19000101|M||2106-3^White^HL70005^^^^2.5.1|12
345 Main St^^San Jose^CA^95125^USA^^^06085||(123)456-7890^PRN^PH^^1^123^4567890||||||||N^Non Hispanic or
Latino^HL70189^^^^2.9||||||||N
ORC|RE|1234d1d1-95fe-462c-8ac6-46728dba581c^Winchester
House^05D2222542^ISO|1234d1d1-95fe-462c-8ac6-46728dba581c^Winchester
House^05D2222542^ISO||||||||1679892871^Doolittle^Doctor^^^^^^CMS&2.16.840.1.113883.3.249&ISO^^^^NPI||(123)456-7
890^WPN^PH^^1^123^4567890|202108020000+0000||||||Winchester House|6789 Main St^^San
Jose^CA^95126^^^^06085|(123)456-7890^WPN^PH^^1^123^4567890|6789 Main St^^San Jose^CA^95126
OBR|1|1234d1d1-95fe-462c-8ac6-46728dba581c^Winchester
House^05D2222542^ISO|1234d1d1-95fe-462c-8ac6-46728dba581c^Winchester
House^05D2222542^ISO|94558-4^SARS-CoV-2 (COVID-19) Ag [Presence] in Respiratory specimen by Rapid
immunoassay^LN^^^^2.68|||202108020000-0500|202108020000-0500|||||||1679892871^Doolittle^Doctor^^^^^^CMS&2.16.
840.1.113883.3.249&ISO^^^^NPI|(123)456-7890^WPN^PH^^1^123^4567890||||202108020000-0500|||F
OBX|1|CWE|94558-4^SARS-CoV-2 (COVID-19) Ag [Presence] in Respiratory specimen by Rapid
immunoassay^LN^^^^2.68||260415000^Not detected^SCT|||N^Normal (applies to non-numeric
results)^HL70078^^^^2.7|||F|||202108020000-0500|05D2222542^ISO||BD Veritor System for Rapid Detection of
SARS-CoV-2_Becton, Dickinson and Company (BD)^BD Veritor System for Rapid Detection of
SARS-CoV-2^99ELR^^^^2.68^^BD Veritor System for Rapid Detection of SARS-CoV-2_Becton, Dickinson and Company
(BD)_EUA||202108020000-0500||||Winchester House^^^^^ISO&2.16.840.1.113883.19.4.6&ISO^XX^^^05D2222542|6789
Main St^^San Jose^CA^95126^^^^06085
OBX|2|CWE|95418-0^Whether patient is employed in a healthcare
setting^LN^^^^2.69||N^No^HL70136|||||F|||202108020000-0500|05D2222542||||202108020000-0500||||Winchester
House^^^^^ISO&2.16.840.1.113883.19.4.6&ISO^XX^^^05D2222542|6789 Main St^^San
Jose^CA^95126-5285^^^^06085||||QST
OBX|3|CWE|95417-2^First test for condition of
interest^LN^^^^2.69||N^No^HL70136|||||F|||202108020000-0500|05D2222542||||202108020000-0500||||Winchester
House^^^^^ISO&2.16.840.1.113883.19.4.6&ISO^XX^^^05D2222542|6789 Main St^^San
Jose^CA^95126-5285^^^^06085||||QST
OBX|4|CWE|95421-4^Resides in a congregate care
setting^LN^^^^2.69||Y^Yes^HL70136|||||F|||202108020000-0500|05D2222542||||202108020000-0500||||Winchester
House^^^^^ISO&2.16.840.1.113883.19.4.6&ISO^XX^^^05D2222542|6789 Main St^^San
Jose^CA^95126-5285^^^^06085||||QST
OBX|5|CWE|95419-8^Has symptoms related to condition of
interest^LN^^^^2.69||N^No^HL70136|||||F|||202108020000-0500|05D2222542||||202108020000-0500||||Winchester
House^^^^^ISO&2.16.840.1.113883.19.4.6&ISO^XX^^^05D2222542|6789 Main St^^San
Jose^CA^95126-5285^^^^06085||||QST
SPM|1|1234d1d1-95fe-462c-8ac6-46728dba581c&&05D2222542&ISO^1234d1d1-95fe-462c-8ac6-46728dba581c&&05D222
2542&ISO||445297001^Swab of internal nose^SCT^^^^2.67||||53342003^Internal nose structure (body
structure)^SCT^^^^2020-09-01||||||||202108020000-0500|20210802000006.0000-0500

Response


```
{
  "id" : "f08ccba0-c5ff-4ef5-924c-f72747603f02",
  "timestamp" : "2021-08-05T11:33:01.060209Z",
  "topic" : "covid-19",
  "reportItemCount" : 1,
  "destinations" : [ {
    "organization" : "California Department of Public Health",
    "organization_id" : "ca-dph",
    "service" : "elr",
    "sending_at" : "2021-08-05T07:34-04:00",
    "itemCount" : 1
  } ],
  "destinationCount" : 1,
  "warningCount" : 0,
  "errorCount" : 0,
  "errors" : [ ],
  "warnings" : [ ]
}
```

Example data models

- COVID-19 data matching HHS Guidance
- A simple schema meant for testing and demos
- A complex real-life schema used by our sister project, SimpleReport, for submitting COVID-19 data
- Other examples of COVID-19 schemas